

# Editorial Días transcurridos

José Santiago Vales Mena

Junio 17, 2024

Aquí está el texto con correcciones de ortografía, gramática y algunas mejoras:

---

## Subtarea 1 [15 puntos]

La primera subtarea indica que los días transcurridos están en el mismo mes y que el primer día es anterior al segundo. Por lo tanto, los días transcurridos son únicamente la resta de los días.

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    ios_base::sync_with_stdio(0);
    cin.tie(0);

    string mes1, mes2;
    int d1, d2;
    cin >> mes1 >> d1 >> mes2 >> d2;
    cout << d2 - d1;
    return 0;
}
```

## Subtarea 2 [15 puntos]

Si el primer mes es julio y el segundo es agosto, basta con ver cuántos días le faltan a julio para terminar y sumarlos a los que han pasado de agosto.

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    ios_base::sync_with_stdio(0);
    cin.tie(0);

    string mes1, mes2;
    int d1, d2;
    cin >> mes1 >> d1 >> mes2 >> d2;
    cout << 31 - d1 + d2;
    return 0;
}
```

## Subtarea 3 [50 puntos]

Digamos que tenemos desde ( m\_1 ) hasta ( m\_2 ). La respuesta se puede ver como la suma de los días de los meses desde ( m\_1 ) hasta el mes anterior a ( m\_2 ), más los días transcurridos en ( m\_2 ) (( d\_2 )) menos los ya

transcurridos en ( d\_1 ). Hay muchas formas de programar esto; aquí se usa un vector de los meses con su cantidad de días.

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    ios_base::sync_with_stdio(0);
    cin.tie(0);

    vector<string> meses = {
        "ENE", "FEB", "MAR", "ABR", "MAY", "JUN", "JUL", "AGO", "SEP", "OCT", "NOV", "DIC"
    };
    vector<int> dias = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

    string mes1, mes2;
    int d1, d2;
    cin >> mes1 >> d1 >> mes2 >> d2;
    pair<int, int> dia1, dia2;

    for (int i = 0; i < 12; i++) {
        if (mes1 == meses[i]) dia1 = {i, d1};
        if (mes2 == meses[i]) dia2 = {i, d2};
    }

    int res = 0;
    res -= d1;
    while (dia1.first != dia2.first) {
        res += dias[dia1.first++];
    }
    res += d2;

    cout << res;

    return 0;
}
```

## Subtarea 4 [100 puntos]

Únicamente debemos hacer lo mismo que en la subtarea 3, recorriendo el calendario (sacando módulo 12) y teniendo cuidado en el caso del mismo mes, pero cuando el primer día es posterior al segundo.

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    ios_base::sync_with_stdio(0);
    cin.tie(0);

    vector<string> meses = {
        "ENE", "FEB", "MAR", "ABR", "MAY", "JUN", "JUL", "AGO", "SEP", "OCT", "NOV", "DIC"
    };
    vector<int> dias = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

    string mes1, mes2;
    int d1, d2;
    cin >> mes1 >> d1 >> mes2 >> d2;
    pair<int, int> dia1, dia2;
```

```

for (int i = 0; i < 12; i++) {
    if (mes1 == meses[i]) dia1 = {i, d1};
    if (mes2 == meses[i]) dia2 = {i, d2};
}

if (mes1 == mes2 && d1 > d2) {
    cout << 365 - (d1 - d2);
    return 0;
}

int res = 0;
res -= d1;
while (dia1.first != dia2.first) {
    res += dias[dia1.first++];
    dia1.first %= 12;
}
res += d2;

cout << res;

return 0;
}

```